
NASA

Table of Contents

Introduction

- **JPL is developing a high fidelity real-time EDL simulator called *DSENDS*.**

- **DSENDS is an extension of the JPL's heritage *DARTS/Dshell*++ Multi-Mission, Real-Time Spacecraft Simulator which has been used by various flight projects.**
  - e.g. Galileo, Cassini, Mars Pathfinder, Stardust, SIM, SRTM, ST-6.

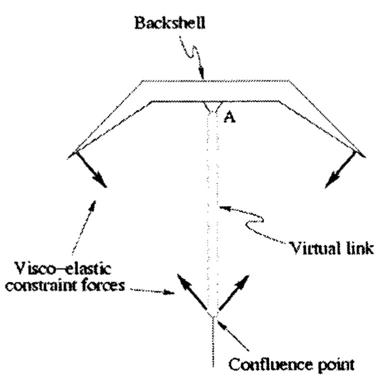
- **Mars Smart Lander has identified *DSENDS* as a supporting real-time simulator.**

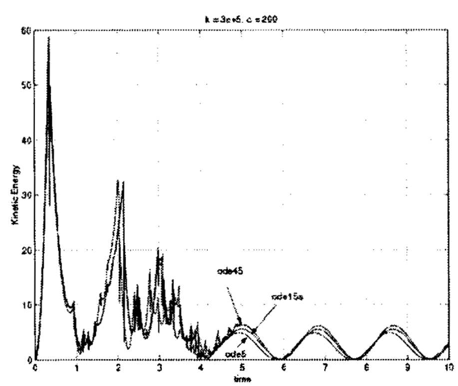
- **EDL-specific interfaces were developed for *DSENDS.***
  - i.e. interfaces to instrument and environmental models.

- **Many systems engineering issues were confronted in the development of the *DSENDS* real-time simulator.**

Real-time Issues In Dynamics Modeling

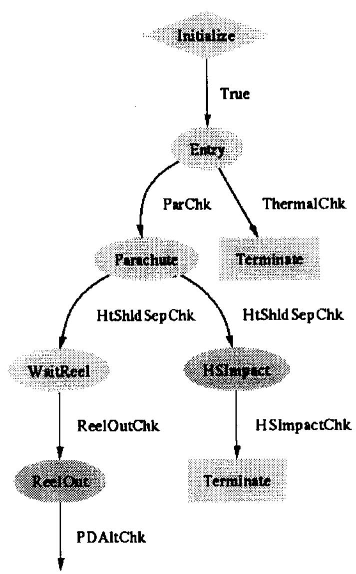
- **Spacecraft dynamics modeling requires an automated method to transition between different configuration and transfer the s/c state between configurations.**
  - *DSENDS* inherited capability to maintain the state of multiple s/c and the capability to "turn on" or "turn off" s/c models.





- **Tether dynamics modeling requires an automated method to decrease the integration step size to capture high frequency dynamics.**
  - Variable step integrators cannot be used since they require an undeterministic number of processor cycles.

Automation Solutions for Dynamics Modeling

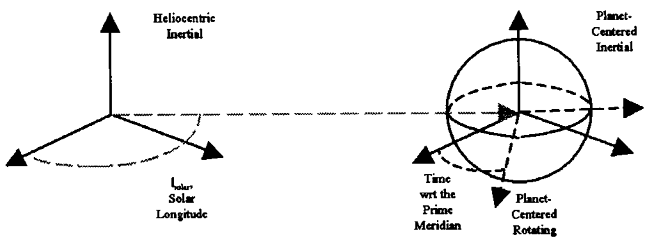


- ***DSENDS State Machine*** **satisfied automation needs.**
  - Provides mechanism to automate execution of user-functions during state execution and state transitions.
  - Allows user-provided functions for testing of state transitions.
  - Manages multiple, simultaneous s/c states.
  - Readily defined and integrated within *DSENDS*.

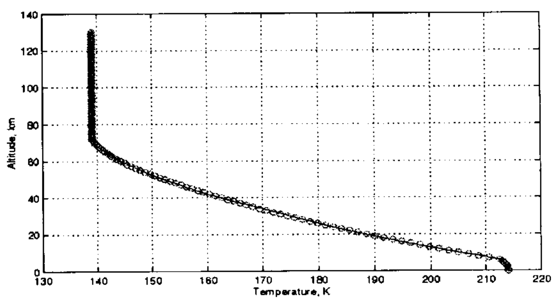
- **Used in spacecraft dynamics modeling.**
  - User-defined transition functions would apply affine transformations between the state of the old configuration and the state of the new configuration.

- **Used in tether dynamics modeling.**
  - User-defined transition functions would shorten the integration step size of the fixed step integrator so that the high frequency dynamics were captured.
  - Another user-defined transition function would length the integration step size when the dynamics were of sufficiently low frequency.

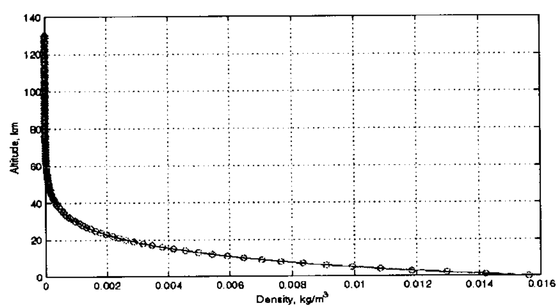
Real-time Issues in Atmospheric Modeling



- **Atmospheric models must vary with:**
  - Planetary inertial position and attitude (i.e. solar long., solar flux, and local time).
  - Planetary surface (i.e. global circulation)
  - Provide atmospheric conditions in a timely manner.

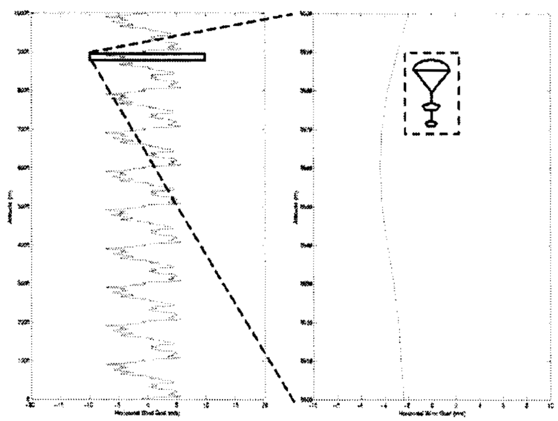


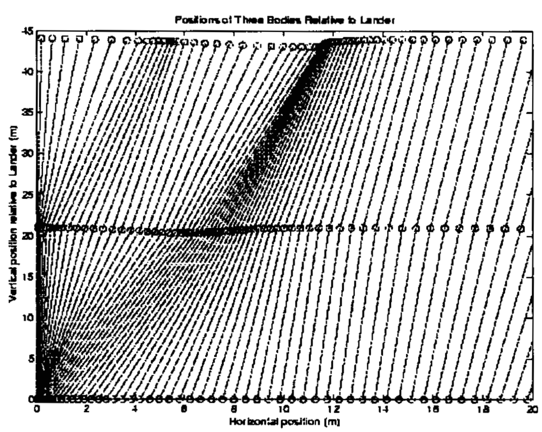
- **Cannot use ODE-driven atmospheric models in real-time simulations.**
  - Likely and unnecessary bottleneck.



- **Interpolated and parametric atmospheric models will behave predictably.**
  - Must quantify processor cycle usage of candidate models.

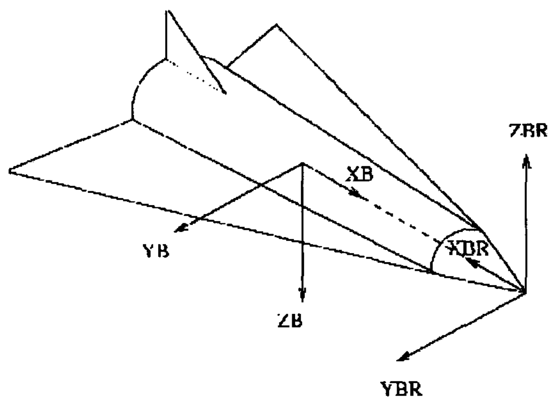
Real-time Issues in Wind Modeling

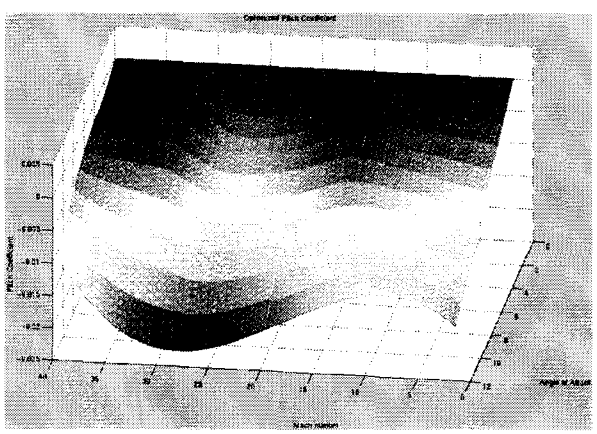




- **Wind models must vary with:**
  - Local time (planetary attitude)
  - Altitude, latitude and longitude.
  - Provide perturbing wind velocities in a timely manner.
- **Cannot use PDE-driven wind models.**
  - Also a likely and unnecessary bottleneck.
- **Some wind effects may be modeled in the atmospheric model**
  - e.g. Mars GRAM has global circulation

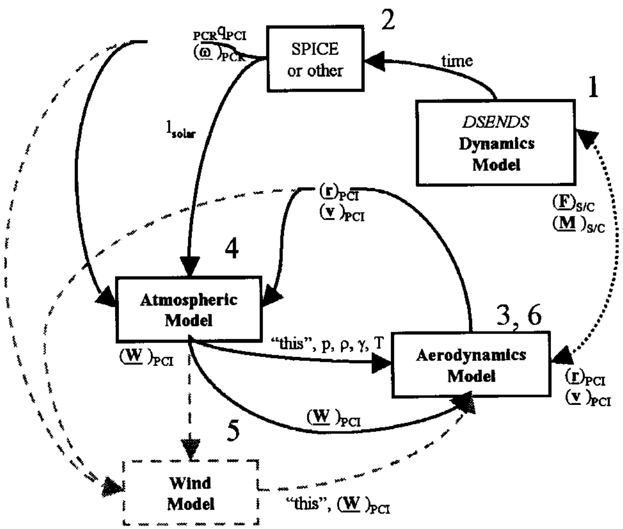
Real-time Issues in Aerodynamics Modeling





- **Aerodynamic Coefficient models must span hypersonic to subsonic velocities.**
  - Most aerodynamics models only operate in a subset of the required velocity domain.
- **Leverage legacy codes (i.e. POST) with high fidelity aerodynamics models.**
  - Must quantify processor cycle usage.
- **Compute aerodynamic forces and moments in a deterministic number of processor cycles.**
  - Total number of processor cycles depends on which atmospheric model and wind model are in use.

DSENDS Aerodynamics Modeling Architecture



- **Decomposes aerodynamics into three tightly coupled models.**
  - Atmospheric model
  - Wind model
  - Aerodynamics model

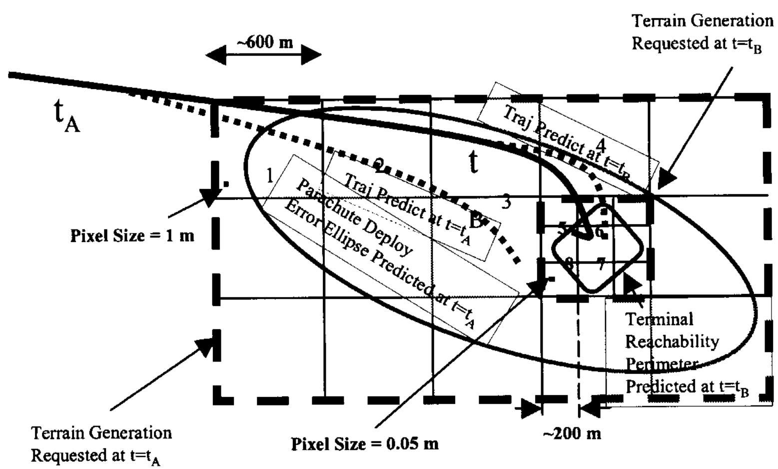
- **Object-oriented model definitions.**
  - Interfaces and functionality of each model are defined so that any model can be overloaded by another.

- **Scale down processor usage by swapping in simpler models.**

- **Aerodynamics model communicates with the other two models and applies aerodynamic forces and moments to the vehicle.**
  - Reports the vehicles inertial state and queries the other two models for atmospheric conditions.

- **Wind models are optional.**
  - Add perturbing wind velocities to wind velocity output of the atmospheric model.

Real-time Issues in Terrain Modeling for Instrument Simulations



Terrain Patches Needed: 1-8

- ***DSENDS* needs terrain in EDL simulations**
  - Compute height over local terrain.
  - Provide terrain observed in an instrument FOV.
  - Inputs for Laser Mapper (LIDAR) model & Phased Array Radar models.
  - Evaluate lander kinematics and dynamics at touchdown.
  - Visual aid.
- **Cannot preload an entire DEM data set at high resolution.**
  - e.g. 200 km by 100 km at 10 cm resolution is 2.0E+14 pixels.
- **Terrain data is not available at high resolution.**
  - May be a data set, synthetic or a convolution of synthetic and observed.
- **"Terrain clients" must have timely updates appropriate for the current EDL system state.**

Instrument Terrain Server

- **Instrument Terrain Server provides required terrain services.**
  - Shared memory communication (i.e. low data transport latencies).
  - Uses several shared memory buffers that contain overlapping terrain segments.
    - When an application requests data in an overlapping segment, buffers are switched in real-time to allow seamless access to terrain data.
  - Uses a predictive model in generating terrain data.
    - Predicts extent and resolution of terrain segments required by the simulation.
  - Uses knowledge of terrain generation times, data transport times and buffer sizes to sequence the generation, transport and uploading of terrain segments.
  - Manages use and re-use of buffers, the extent of overlap and some level of cache management to relieve the simulation from frequent transactions with the terrain generation and transport processes.
    - Terrain generation takes many seconds while the transport and buffer management process consume only fractions of seconds.

Summary

- **Real-time issues were considered for:**
  - Spacecraft and tether dynamics
  - Aerodynamics
  - Terrain services
- **Have designed and implemented solutions**
  - *DSENDS* State Machine
  - *DSENDS* Aerodynamics Modeling Architecture
  - Instrument Terrain Server

Future Work

- **Quantify performance of *DSENDS* components.**
  - Needed for a processing budget

- **Populate *DSENDS* with models.**
  - Spacecraft models (as architecture changes).
  - Atmospheric models.
  - Wind models.
  - Aerodynamics models.

- **Validate and verify *DSENDS* models.**

Backup Slides

Spacecraft Dynamics Modeling

- **Significant challenge to manage transitions between different s/c configurations during EDL.**
  - e.g. heat shield separation, parachute deployment, etc.

- **Required automated methods of applying affine transformations between the s/c state in an old configuration to s/c state in a new configuration.**
  - How to transfer rotational, translational and positional information between s/c configurations.

- **Required automated methods for maintaining continuity and accuracy across transitions.**
  - Insure that high frequency transients are captured by the simulator's integration scheme (i.e. temporarily decrease the integration step size).

Tether Dynamics Modeling

- **Flight train is expected to have a triple bridle system that connects the reeled-out lander to the backshell.**
  - Similar to Mars Pathfinder flight train.
  - Three flexible tether lines attach to the backshell and combine at one end to make a single confluence point.

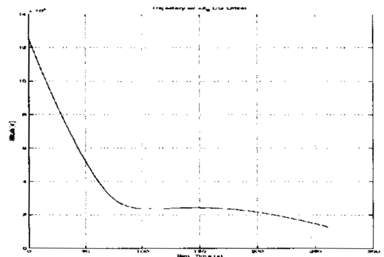
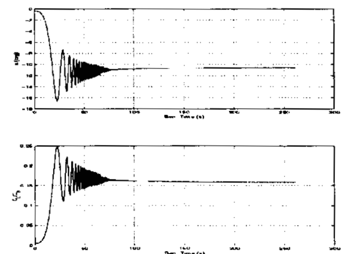
- **Each tether line requires a visco-elastic model and all the tethers are coupled via single-ended constraint equations.**
  - Must capture the motion of the confluence point as the tether lines stretch, contract and go slack.

- **Exhibits high frequency dynamics when the lander is initially released from the backshell.**
  - Requires the simulator have small integration steps until the flight train system reaches a quasi-equilibrium with no high frequency dynamics.

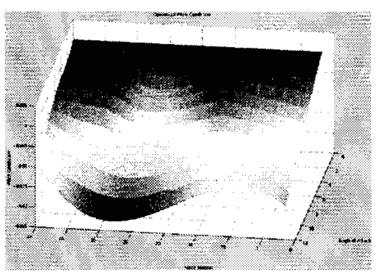
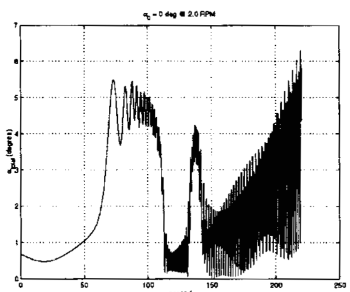
DSENDS Aerodynamics Modeling

Aero. Coef. Models at different fidelity

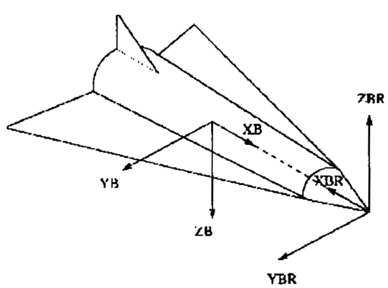
- Linearized Aero. Coefficient Model

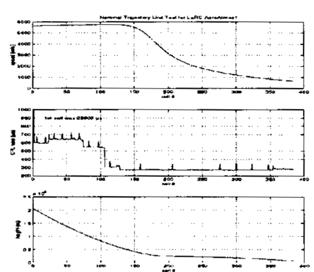



- Interpolated Aero. Coefficient Model

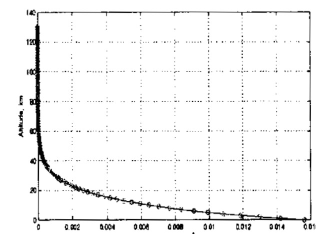
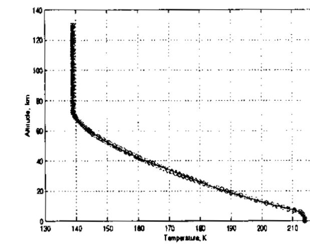
- LaRC Hypersonic Entry Model





Atmospheric Models

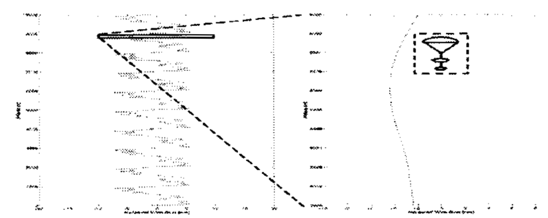
- Fitted Atmospheric Profiles

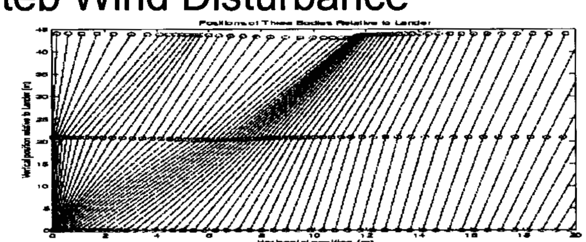
- Mars GRAM (almost there)

Testing Wind Models on Flight Train

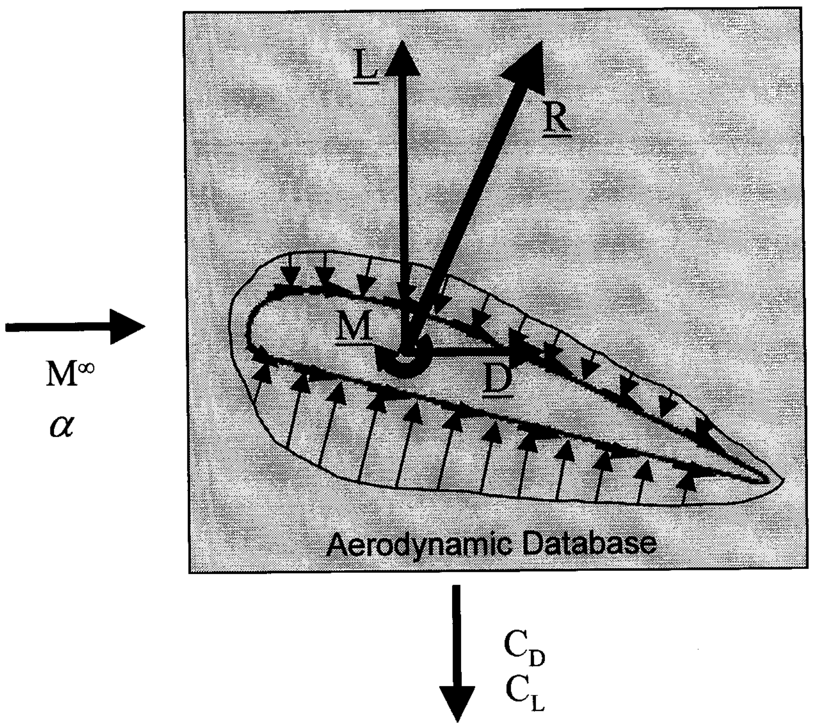
- Chia-Yen Peng's Wind Model



- Step Wind Disturbance



Aerodynamic Databases



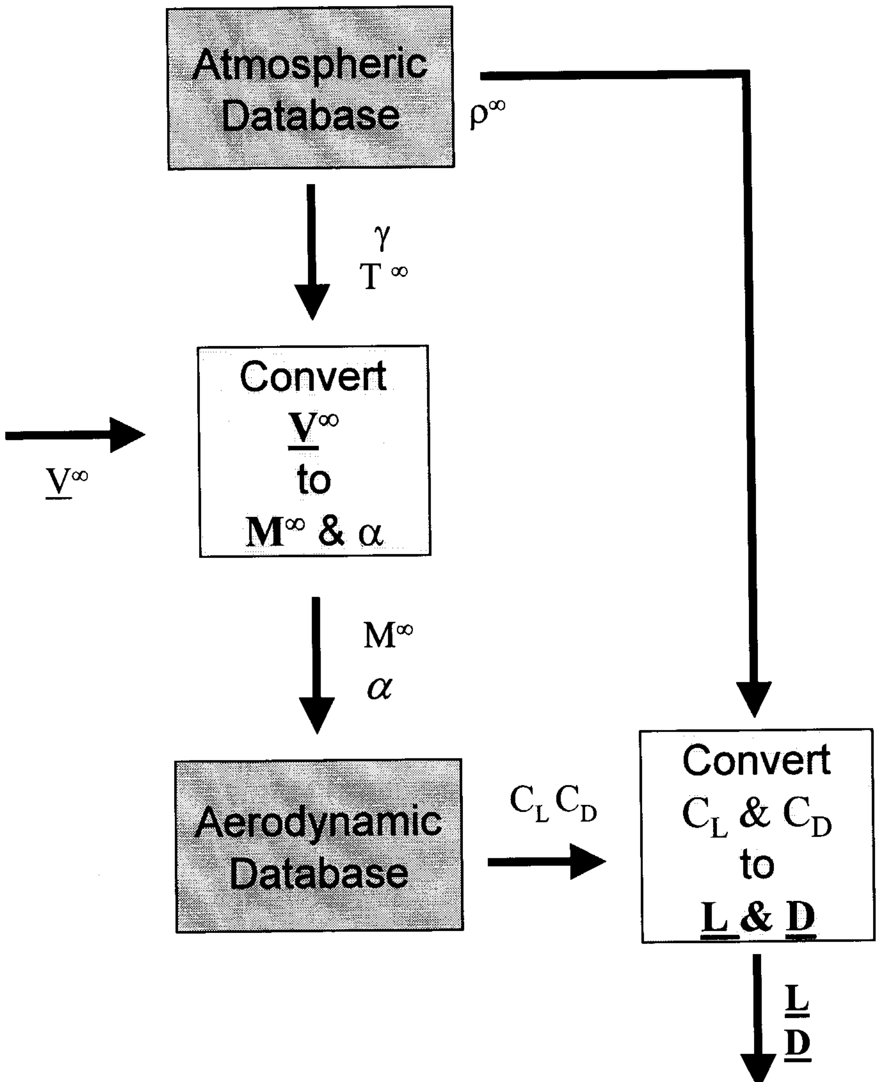
Aerodynamic force & **moment** can be tabulated for different freestream and atmospheric conditions.

Typically, the freestream velocity and the **aerodynamic force** & **moment** are converted into dimensionless quantities:

- Freestream Mach Number, $M^{\infty} = \dfrac{\left|\vec{V}^{\infty}\right|}{\sqrt{\gamma RT}}$
- Angle of attack, α
- Coefficient of drag, $C_D = \dfrac{\left|\vec{D}\right|}{0.5\rho^{\infty}\left|\vec{V}\right|^2}$
- Coefficient of lift, $C_L = \dfrac{\left|\vec{L}\right|}{0.5\rho^{\infty}\left|\vec{V}\right|^2}$
- etc.

Database entries acquired from CFD, DSMC or wind tunnel results. **Database limited by its sampling and interpolation scheme**.

Atmospheric Databases



Atmospheric databases are necessary for utilizing an **aerodynamic database**.

The **atmospheric database** is used to convert the freestream conditions into the dimensionless quantities stored in the **aerodynamic database**.

Once the lift, drag, etc. coefficients are extracted, the **atmospheric database** is used to convert those coefficients into **aerodynamic force** & **moment**.

Database limited by its sampling and interpolation scheme.

Wind Modeling

$$\left(\vec{V}^{\infty}\right)_A = \left(\vec{V}^{S/C}\right)_A + \underbrace{\left(\vec{\Omega}\times\vec{r}\right)_A + \left(\vec{W}\right)_A}_{\text{Steady}} + \underbrace{\left(\vec{w}\right)_A}_{\text{Variable}}$$

- Freestream velocity **is the sum of the S/C and wind velocities**

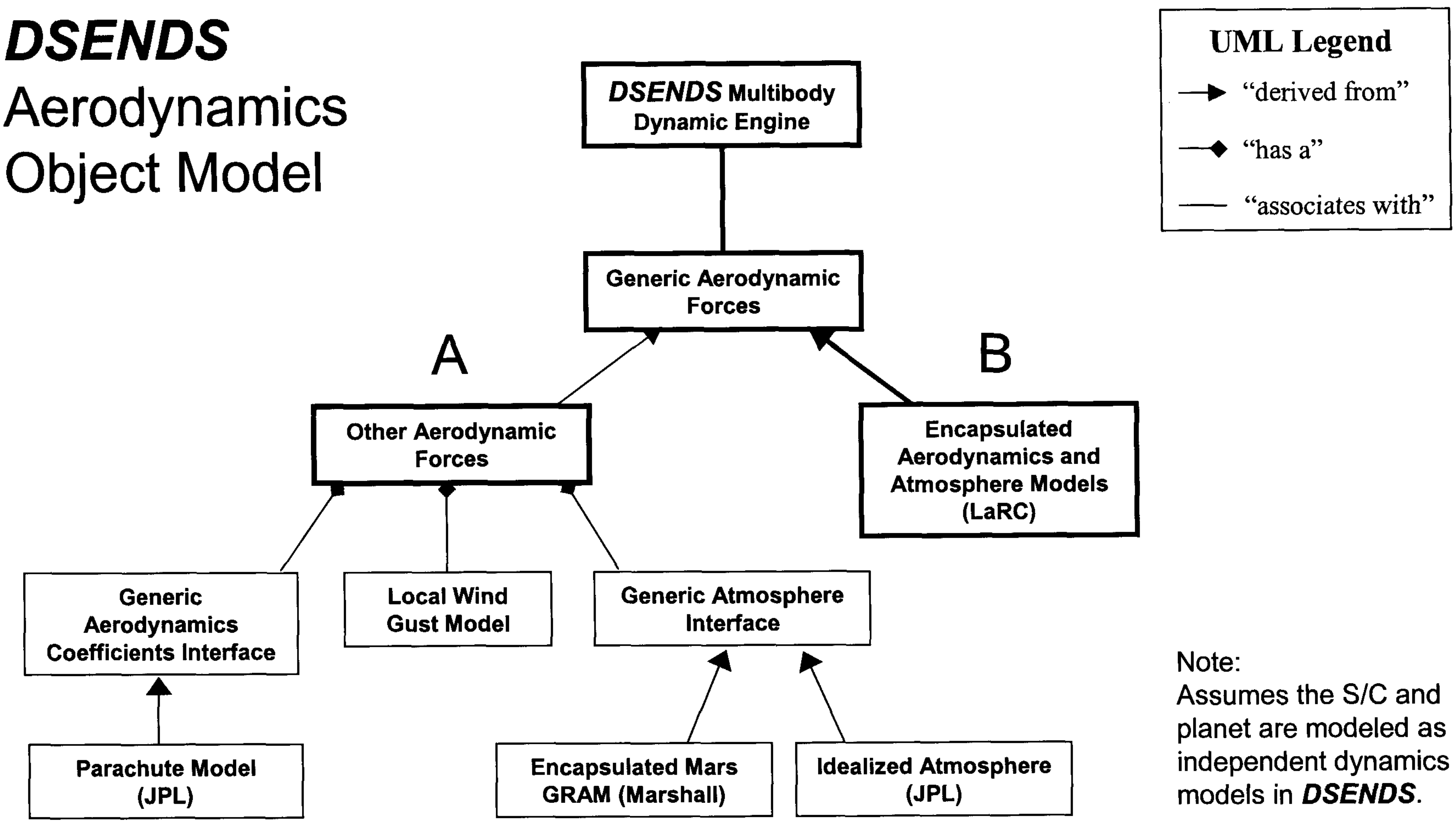
- **Wind velocity has steady global and variable local components**
  - Steady Global Wind Velocities
    - Planetary Rotation $(\underline{\Omega} \times \underline{r})_A$
    - Global Wind Velocity (i.e. atmospheric circulation) $(\underline{\mathbf{W}})_A$
  - Variable Local Wind Velocities
    - Local Wind Gusts (i.e. atmospheric circulation) $(\underline{\mathbf{w}})_A$

- **Local wind gusts can be modeled as a zero mean, Gaussian process**
  - e.g. MER/MPF used Earth variable wind PSD since there is no Martian PSD

$$G_u(f) = (2.4\times10^{-4})f^{2.4} + 0.01f^{-5/3}$$

OO Aerodynamics Model

DSENDS
Aerodynamics
Object Model



Note:
Assumes the S/C and planet are modeled as independent dynamics models in ***DSENDS***.

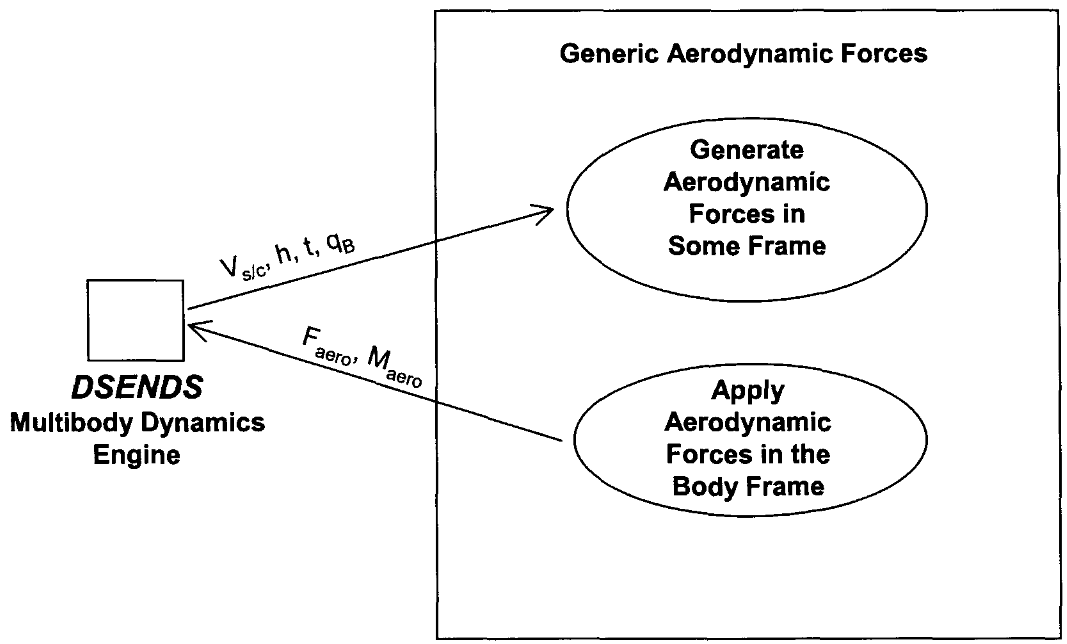
OO Aerodynamics Model

DSENDS Aerodynamics Use Case

UML Legend

→ input or output

◯ capability



Note: The above describes a "black box."